Discussion Paper No. 592

THE SLOW SERVER PROBLEM: A QUEUE WITH STALLING

by

Michael Rubinovitch*

April 1984

*Department of Managerial Economics and Decision Sciences, Northwestern University, Evanston, Illinois 60201. Currently on leave from the Technion, Israel Institute of Technology.

Abstract

A queue with Poisson arrivals and two different exponential servers is considered. It is assumed that customers are allowed to stall, i.e., to wait for a busy fast server at times when the slow server is free. A stochastic analysis of the queue is given, steady state probabilities are computed, and policies for overall optimization are characterized and computed. The issue of individual customer's optimization versus overall optimization is also discussed.

THE SLOW SERVER PROBLEM: A QUEUE WITH STALLING

by
Michael Rubinovitch

Introduction

This is a study of queuing systems with Poisson arrivals and two different exponential servers. Server 1 is the fast server and Server 2 is the slow server (in the sense of mean service times). The problem is to find the best operating policy so as to minimize the mean sojourn time of customers in the system.

In Rubinovitch [1983] it was assumed that the controller of this system has only two options. He can either always make the slow server available for rendering service or not use it at all. It was shown that the optimal policy is characterized by a single critical number. When traffic intensity exceeds this number both servers should be used. Otherwise, the slow server should be removed and never used. The critical number depends on the ratio of mean service times of the two servers and on customers' behavior. Three cases were considered: (i) customers who arrive to an empty system choose their server at random; (ii) customers who arrive to an empty system always join the fast server; and (iii) customers who arrive to an empty system join the fast server with probability p and the slow server with probability (1 - p). (i) is the case when customers are uninformed; (ii) is the case of informed customers; and (iii) is the case of a partially informed population of customers. The critical number is smallest in (i), largerst in (ii) and increases with p in (iii).

In this paper the same problem is considered except that customers are allowed to stall, i.e., to wait for a busy fast server at times when the slow server is free.

Let λ be the arrival rate and μ_1, μ_2 be the service rates of Server 1 and Server 2, respectively ($\mu_2 < \mu_1$). Let $r = \mu_2/\mu_1$ and consider the case when customers are fully informed and can choose their server. Then an arriving customer who finds a free fast server will join it. If he finds a busy fast server, an idle slow server and j customers waiting for the fast server he will join the slow server whenever $j + 1 > 1/r$, will prefer to join those waiting for the fast server if $j + 1 < 1/r$, and will be indifferent if $j + 1 = 1/r$. Following this rule he will minimize his expected time in the system. Thus, if M is the integer part of $1/r$, then customers do not join the slow server as long as the number in the system is M or less. As can be seen, the decision problem of individual customers is simple (although the queuing process that optimal customer behavior gives rise to is complex).

On the other hand, the problem of overall optimization, i.e., of assigning customers to servers so as to minimize the mean time in the system over all customers is more interesting, more difficult, and also of more practical importance. It is the problem of interest wherever "customers" are inanimate--for example in computer systems or data communications networks where "customers" are jobs to be processed or messages to be transmitted, "servers" are processors or communication channels, and the space for waiting is a buffer. Then there is usually a controller that monitors the number of jobs in the buffer and, depending on the state of system, assigns jobs to processors.

Lin and Kumar [1982] provided a proof that the overall optimal policy is of the threshold type--that is, of the same type as the optimal policy that individual customers follow but presumably with a different M. They use policy iterations on the discounted cost problem and then take the average cost (mean waiting time) problem as the limit. Warland [1983] gave a proof

for the same result using a coupling argument. In any case, irrespective of whether it is individual customers optimizing their welfare or a central agency that assigns customers to servers in a socially optimal way, the queuing process is of the same type. A number K is specified and customers enter the slow server only when the number in the system is K + 1 or more. The term queues with stalling seems an appropriate name for such systems.

In this paper we provide a stochastic analysis for a two-server Markovian queue with stalling. We show how the underlying stochastic processes can be analyzed without solving any new problem, but rather by appealing to known results on processes whose structure is well understood. With this the steady state probabilities, the optimal threshold levels, and other system characteristics can be readily computed. The present analysis can also be used to solve a hierarchic system of several servers, each with its own buffer, where the input to a buffer is the overflow from the buffer above it in the hierarchy. This will be taken up in a separate publication.

In Section 1 we outline the main ideas behind the present appraoch. Details are given in sections 2 and 3. The former section studies a modified (loss) system which may be of indepedent interest, and the latter provides the results for the queue with stalling. Section 4 is a short discussion of issues regarding social optimization and individual customers' optimization.

1. Outline of the Analysis

Let a number K be specified and consider the queue with stalling in which customers join the slow server when the number in system in K + 1 or more. A good way to visualize this is to think of a two-buffer system as shown in Figure 1. Buffer S is of size K - 1 and Buffer Q is of unlimited capacity. As customers come in they join Server 1 if they can. If this server is busy they are placed in Buffer S and if this buffer is full they try to enter

Server 2. If the latter is also busy they are placed in Buffer Q. Buffer S feeds Server 1 only (the customers in it are committed to Server 1). Buffer Q feeds both S and Server 2, whichever can first accept the head-of-the-line customer in Q.

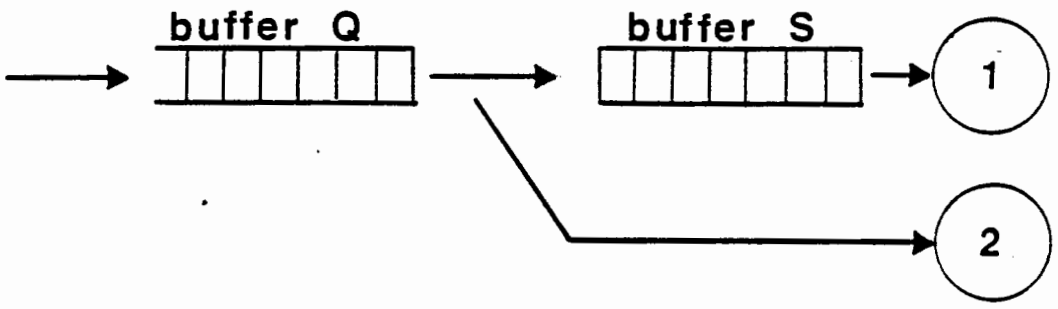

Figure 1. Buffered Queue with Stalling

Let X_t^1 be the number of customers that, at time t, are in Buffer Q, or in Buffer S, or in service with Server 1. Let X_t^2 be the number of customers with Server 2 at time t so that $X_t^1 + X_t^2$ is the total number of customers in the system. Then $X = (X_t^1, X_t^2)_{t>0}$ is a Markov process on $\{0,1,2,\ldots\} \times \{0,1\}$. A direct derivation of steady-state probabilities by solving the steady-state equations is of course possible but will unavoidably be lengthy and complicated (see Larsen [1981] where a different state space is used). The following stochastic analysis is simpler and yields more information on the underlying processes. It is based on the following ideas. First is the observation that it is enough to solve the modified loss system which ceases to accept new arrivals when there are K + 1 customers in the system. Let $\hat{X} = (\hat{X}_t^1, \hat{X}_t^2)_{(t>0)}$ be the same as X but for the modified system (i.e.,

under the assumption that X has a reflecting barrier at state (K,1)). Then note that $\hat{X}^1$ is the same as the number of customers in an M/M/1/K loss system that is simple and well understood. Finally, observe that in the modified system the input to Server 2 is a renewal process so that $\hat{X}^2$ is a GI/M/1/1 queuing process. In fact, $\hat{X}^2$ is the same as the process describing the "locked"/"unlocked" states of type I counter for which all the necessary results are readily available. With these we can completely analyze the modified and the original queuing processes.

2. The Modified (Loss) System

The modified system is as shown in Figure 1 exept that buffer Q is removed and customers arriving when the system is full are lost. This may be of independent interest.

We use the following notation:

$$\mu = \mu_1 + \mu_2,$$

$$r = \mu_2/\mu_1 \qquad (0 < r < 1),$$

$$\rho = \lambda/\mu,$$

$$\rho_1 = \lambda/\mu_1 = (1 + r)\rho$$

Consider first the input to Server 2. Let $Z_t = I_{\{\hat{X}^1_t = K\}}$ and note that Z is an alternating renewal process which stays at state 1 for exponential intervals (mean μ_1^{-1}). The input to Server 2 is a Poisson process which is turned on when Z = 1 and off when Z = 0. Thus, successive times of arrival to Server 2 form a renewal process. Let N_t be the number of renewals in (0,t] and $U(t) = E[N_t]$. Suppose that $Z_0 = 1$ and let $R_1, R_2, \ldots$ be the lengths of successive intervals during which $Z_t = 0$. Then by a straightforward renewal

argument

$$U(t) = \lambda(1 - e^{-\mu_1 t})/\mu_1 + \int_0^t \mu_1 e^{-\mu_1 s}\, ds \int_0^{t-s} P\{(R_1 \in du\}\, U(t - s - u)$$

$$(1) \quad U^*(\theta) = \int_0^\infty e^{-\theta t} U(dt) = \lambda[\mu_1 + \theta - \mu_1 C_K(\theta)]^{-1}$$

where

$$(2) \qquad C_K(\theta) = E[e^{-\theta R_1}].$$

To evaluate $C_K(\theta)$ note that R_1 is the same as the first passage time from state K - 1 to state K in the process $\hat{X}^1$ which, in turn, is the same as the first passage time from state K - 1 to state K in an M/M/1 queue with parameters λ and μ_1. Let T_K be this first passage time. Then $C_K(\theta) = E[e^{-\theta T_K}]$ is given in Bailey [1957] in terms of the roots of a quadratic equation. The following recursive relation can provide an easy proof for Baily's old result (see Appendix) and is more suitable for numerical computations which we later do. Clearly, T_1 is an exponential random variable with mean μ_1^{-1}. Hence

$$C_1(\theta) = \lambda/(\lambda + \theta).$$

Then using again a renewal argument

$$(3) \qquad C_K(\theta) = \lambda/(\lambda + \mu_1 + \theta) + \mu_1 C_{K-1}(\theta) C_K(\theta)/(\lambda + \mu_1 + \theta) \qquad (K > 2)$$

Thus,

$$C_K(\theta) = \lambda[\lambda + \mu_1 + \theta - \mu_1 C_{K-1}(\theta)]^{-1}. \tag{4}$$

We shall also need the mean interarrival time to Server 2. For this, take derivatives in (3), let $\theta \to 0$ and then, by induction, obtain

$$E[T_K] = \begin{cases} \dfrac{1 - \rho_1^K}{\mu_1 \rho_1^K (1 - \rho_1)} & \rho_1 \neq 1, \\ K/\mu_1 & \rho_1 = 1. \end{cases} \tag{5}$$

Let

$$\hat{b}_i(t) = P\{\hat{X}_t^2 = i\} \qquad (i = 1,2)$$

$$\hat{b}_i = \lim_{t \to \infty} \hat{b}_i(t).$$

The limit always exists and b_i is the steady state probability that the slow server is in state i. Also $\hat{b}_i$ are the state probabilities of a type I counter with arrivals according to U and locking time exponential with mean μ^{-1}. This is given by

$$\hat{b}_1 = \frac{1}{m\mu_2}[1 + \int_0^\infty \mu_2 e^{-\mu_2 t} U(t)dt]^{-1}, \tag{6}$$

where m is the mean interarrival time to the counter (See Prabhu [1965a], page 180). It can be obtained directly from (1) but this is not necessary since here interarrival times have the same distribution as T_{K+1} the first passage time form state K to state K + 1 in an M/M/1 system. Thus, using (5) and (6)

we have

$$(7)\qquad \hat{s}_1 = \begin{cases} \dfrac{\rho_1^{K+1}(1-\rho_1)(1+r-C_K(\mu_2))}{r(1-\rho_1^{K+1})((1+r)(1+\rho)-C_K(\mu_2))} & \rho_1 \neq 1, \\ \dfrac{1+r-C_K(\mu_2)}{r[(1+r)(1+\rho)-C_K(\mu_2)]} & \rho_1 = 1. \end{cases}$$

Again, the number $C_K(\mu_2)$ can be computed recursively using (4) or using the explicit expression in the Appendix. Let

$$\hat{\pi}_{ij}(t) = P\{\hat{X}_t^1 = i,\ \hat{X}_t^2 = j\}$$

$$\hat{a}_i(t) = P\{\hat{X}_t^1 = i\}$$

$$\hat{\pi}_{ij} = \lim_{t\to\infty} \hat{\pi}_{ij}(t), \quad \hat{a}_i = \lim_{t\to\infty} \hat{a}_i(t)$$

and note that

$$(8)\qquad \hat{a}_i = \hat{\pi}_{i0} + \hat{\pi}_{i1}.$$

Since $\hat{a}_i$ are the steady state probabilities of an M/M/1/K system we immediately have

$$(9)\qquad \hat{a}_i = \frac{(1-\rho_1)\rho_1^i}{1-\rho_1^{K+1}} \qquad i = 0,1,\ldots,K$$

see, for example, Prabhu [1965b]. Also, is is not difficult to check that

$$\frac{d}{dt}\hat{b}_1(t) = -\mu_2\hat{b}_1(t) + \lambda\hat{\pi}_{K0}(t),$$

so

$$\hat{\pi}_{K0} = \lambda \hat{b}_1/\mu_2 \tag{10}$$

which, together with (8) for i = K gives

$$\hat{\pi}_{K1} = \frac{(1 + r)\rho \hat{a}_K - \hat{b}_1 r}{(1 + r)\rho} \tag{11}$$

We thus have explicit expressions for the following steady-state probabilitiies: $\hat{b}_1(\hat{b}_0)$, the probability that Server 2 is busy (idle); $\hat{a}_i$, the probability that there are i customers committed to Server 1; and $\hat{\pi}_{K1}$ the probability that the system is blocked. The latter is the "loss formula" (11) for a two-server system with stalling and no waiting space. From here one can proceed recursively in a straightforward manner to compute all steady-state probabilities from the steady-state equations:

$$\hat{\pi}_{01}(\lambda + \mu_2) = \mu_1 \hat{\pi}_1, \tag{12a}$$

$$\hat{\pi}_{ij}(\lambda + \mu) - \lambda \hat{\pi}_{i-1}j + \mu_1 \hat{\pi}_{i+1}j \qquad 1 \leq i \leq K-1 \tag{12b}$$

$$\hat{\pi}_{K1}\mu = \lambda \hat{\pi}_{K0} + \lambda \hat{\pi}_{K1}. \tag{12c}$$

Such a computation would start with (12c) and work backwards to (12a) using along the way (8) and (9). We will not pursue this here. Note that since $\hat{b}_i(t)$ are known (see Prabhu [1965b]), a complete time dependent analysis of X can be carried through if it is of interest. Here we need only the mean number of customers in the system

$$\hat{L}_K = \sum_{n=0}^{k} n\hat{a}_n + \hat{b}_1,$$

where $\hat{b}_1$ is known and the first term on the right hand side is the mean number of customers in an M/M/1/K system, with arrival rate λ and service rate μ. Hence

$$\hat{L}_K = \frac{\rho_1[1 - K\rho_1^K(1 - \rho_1) + \rho_1^K]}{(1 - \rho_1)(1 - \rho_1^{K+1})} + \hat{b}_1. \tag{13}$$

3. The Queue With Stalling--Optimal Policies

Consider now the system with stalling and its underlying queueing process $X = (X_t^1, X_t^2)_{t>0}$. We use the same notation as in the modified system except that the "hats" are removed. Assume that $\rho < 1$ so that a steady-state distribution exists and we first compute the steady state probabilities.

For this, note that the transition probabilities of X and $\hat{X}$ are the same on $\{0,1,\ldots,K\} \times \{0,1\}$, while for $i \geq 1$ we have

$$a_{K+i} = \rho^i \pi_{K1}. \tag{14}$$

It follows that there exists a number, say, α_K, such that

$$\pi_{ij} = \alpha_K \hat{\pi}_{ij}, \tag{15}$$

$$a_i = \pi_{i0} + \pi_{i1} = \alpha_K \hat{a}_i \tag{16}$$

for $i = 1,\ldots,k$, $j = 0,1$. Furthermore, from (14) and (16)

$$1 = \sum_{i=0}^{\infty} a_i = \alpha_K \sum_{i=0}^{K} \hat{a}_i + \alpha_K \sum_{i=K+1}^{\infty} \rho^i \hat{\pi}_{K1}$$

so

$$\alpha_K = \frac{1 - \rho}{1 - \rho + \rho\hat{\pi}_{K1}}. \tag{17}$$

Since $\hat{\pi}_{K1}$ is known (see (11), (7) and (9)) one can compute all the steady-state probabilities using (14) and (15) together with (12) and (17). In particular, the mean number of stalling customers is $\alpha_K(\hat{L}_K - \hat{b}_1)$ and the fraction of time Server 2 is busy is

$$b_1 = \alpha_K(\hat{b}_1 + \frac{\hat{\pi}_{K1}\rho}{1 - \rho}). \tag{18}$$

The mean number of customers in the system is

$$\begin{aligned} L_K &= \sum_{i=0}^{\infty} if_i + b_1 \\ &= \sum_{i=0}^{K} if_i + \sum_{i=K+1}^{\infty} if_i + b_1 \\ &= \alpha_K(\hat{L}_K + \frac{\rho\hat{\pi}_{K1}[(K + 1)(1 - \rho) + 1]}{(1 - \rho)^2}) \end{aligned} \tag{19}$$

Other system characteristics can also be evaluated. For example, the distribution of idle and busy periods for Server 2 can be computed directly from $b_1(t)$ which is known since $\hat{b}_1(t)$ is known (see Prabhu [1965a]). Also note that exactly the same analysis as given here applies to a loss system with stalling--i.e., for the system shown in Figure 1 except that buffer Q is finite and arrivals which occur when buffer Q is full are lost. All the results given above apply to this system except that that state space is

finite and α_K is given by

$$\alpha_K^{(N)} = \frac{1 - \rho}{1 - \rho + \rho\hat{\pi}_{K1}(1 - \rho^{N+1})}$$

where N is the size of buffer Q. The loss formula for this case is the expression of (11) with $\alpha_K^{(N)}$ replacing α_K.

Turning now to computations of optimal policies, let $K_0(r,\rho)$ be the optimal number of stalling customers when the objective is to minimize the mean time spent in the system and r and ρ are specified. In other words, $K_0(r,\rho)$ is the optimal size of buffer S. This optimal number can be computed along the lines of Rubinovitch [1983] by chacterizing the region in the (r,ρ) plane for which $L_K < L_{K+1}$ and repeating this for each K. This involves messy algebraic work, the outcome of which would be in terms of polynomial inequalities in ρ that can be solved only numerically. Thus, a direct computational approach is in order as follows.

Since for each set of values for r, ρ, and K one can readily compute the value of L_K, it is easy to develop a search procedure which will find for each r the critical value of ρ below which $L_K < L_{K+1}$ and above which $L_K > L_{K+1}$. Such a computation was carried out and the results are given in Figure 2. It shows for K = 0,...,9, the boundaries of the regions in which $K_0 = K_0(r,\rho)$ is optimal. As we see, K_0 is decreasing in r for fixed ρ, and in ρ for fixed r, as it should. Furthermore, as $\rho \to 0$ the optimal K_0 becomes the same as the optimal number of stallers when customers are allowed to pursue self optimization (see section 4). On the other hand, as $\rho \to 1$ there is a sequence of numbers, say, $r_1^*, r_2^*, \ldots$, with the property that $L_K > L_{K+1}$ when $r > r_K^*$ and $L_K < L_{K+1}$ when $r < r_K^*$. This author was unable to derive explicit expressions for these numbers.

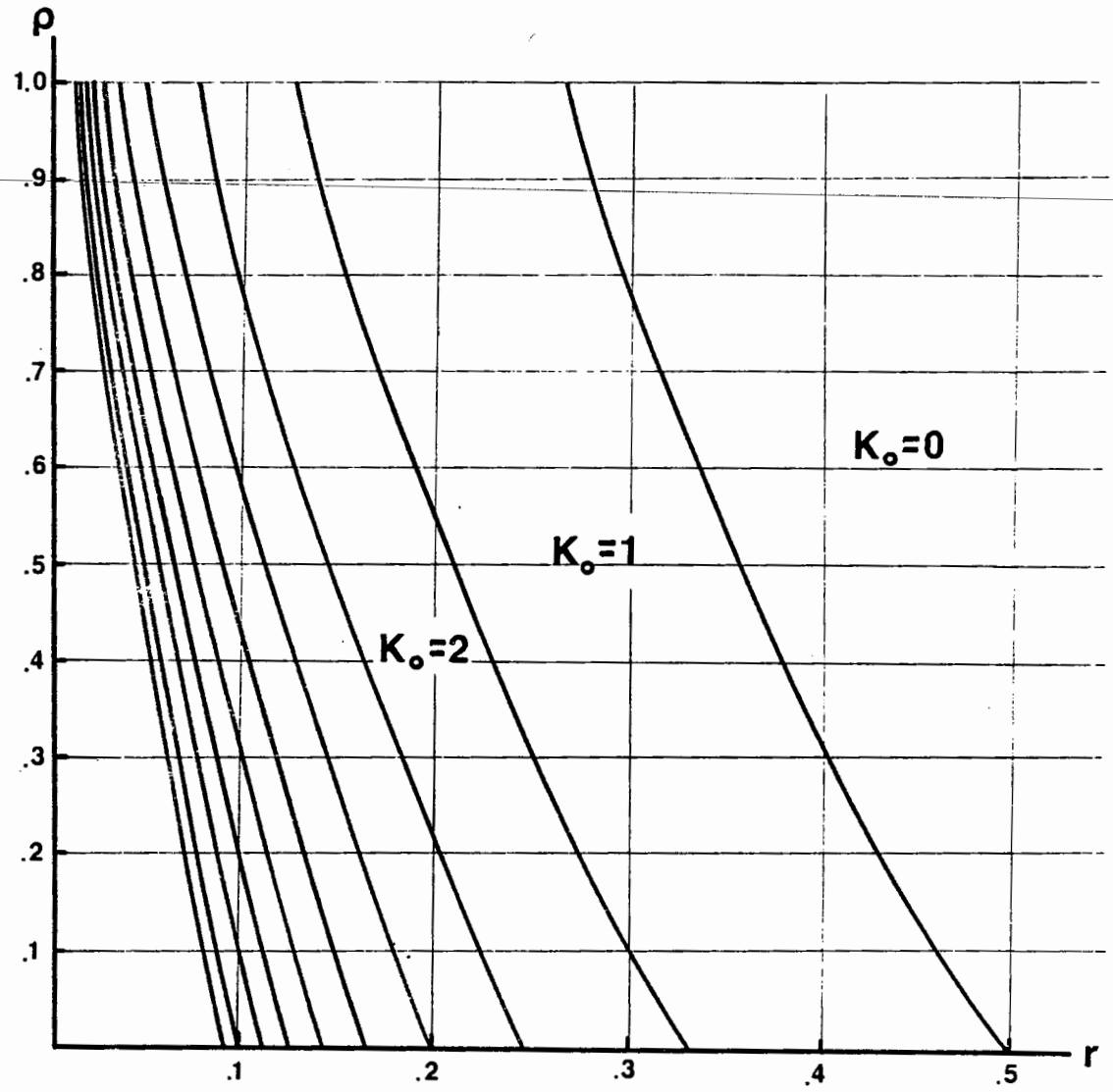

Figure 2. Optimal Policies

Having obtained the optimal operating policies it is of interest to compare the performance of the otpimal system with stalling to the performance of systems without stalling. Tables 1 and 2 show computational results for this comparison. For each ρ and each r the first column is the optimal threshold level $K_0(r,\rho)$, i.e., the optimal size of buffer S. The second

column, L_{K_0}, is the mean queue length of this optimal system. The third column (column (1)) is the percentage reduction in mean queue length when one uses the optimal system with stalling instead of the system with two servers without stalling. The entries in this column are $100\ (L_0 - L_{K_0})/L_0$. The fourth colum (2) is the percentage reduction in queue length when the system with K_0 is used instead of the system which employs the fast server only. The entries in this column are $100[\rho_1/(1 - \rho_1) - L_{K_0}]/(\rho_1/(1 - \rho_1))]$. Note that when $\rho_1 = (1 + r)\rho > 1$ the system with the fast server only is saturated and this is indicated by the symbol ∞ in column (2).

Several interesting conclusions can be drawn from the data in Table 1 and Table 2. First is the observation that while the system with K_0 is always better than the two other systems, the difference is substantial only for a range of intermediate values of ρ. For small ρ the system with K_0 performs about the same as the system with the fast server only, while for large ρ its performance is about the same as the two server system without stalling. The intuition behind this is clear since when ρ is small the system with K_0 rarely uses the slow server and when ρ is large it uses the slow server at almost all times. In any case, in practice one would probabily prefer to use the sytem without stalling when ρ is large, the system with the fast server only when ρ is small and the optimal system with stalling when ρ is in an intermediate range of values. This range depends, of course, on r. For example, when $r = 0.05$ it could be from $\rho = 0.70$ to $\rho = 0.90$. For $r = 0.3$ it could be from $\rho = 0.25$ to $\rho = 0.50$.

Another interesting observation is that the performance of the optimal system with stalling does not change much as r changes while ρ remains the same. In fact, it may be surprising that even when r is small, substantial savings can be achieved by employing the slow server and allow stalling. For

Table 1. Optimal Threshold Levels and System Performance.

	r = 0.05				r = 0.10				r = 0.15			
ρ	K_0	L_{K_0}	(1)	(2)	K_0	L_{K_0}	(1)	(2)	K_0	L_{K_0}	(1)	(2)
.05	14	.06	43.6	0.0	7	.06	26.7	0.0	5	.06	18.1	0.0
.10	14	.12	54.4	0.0	7	.12	36.9	0.0	5	.13	26.1	0.0
.15	14	.19	57.2	0.0	7	.20	40.8	0.0	4	.21	29.5	0.0
.20	14	.27	56.9	0.0	7	.28	41.5	0.0	4	.30	30.2	0.0
.25	14	.36	55.0	0.0	6	.38	40.5	0.0	4	.40	29.2	.1
.30	13	.46	52.1	0.0	6	.49	38.3	0.0	3	.52	27.2	.5
.35	12	.58	48.6	0.0	5	.63	35.3	.1	3	.67	24.5	1.2
.40	11	.72	44.7	0.0	5	.78	31.7	.4	3	.83	21.4	2.6
.45	10	.90	40.3	0.0	4	.97	27.7	1.0	3	1.02	18.2	4.6
.50	9	1.10	35.6	.1	4	1.19	23.6	2.3	2	1.25	15.1	7.7
.55	8	1.36	30.7	.3	4	1.46	19.5	4.4	2	1.51	12.4	12.0
.60	7	1.69	25.5	.8	3	1.79	15.7	7.6	2	1.83	9.9	17.6
.65	7	2.10	20.4	2.1	3	2.20	12.4	12.3	2	2.23	7.8	24.6
.70	6	2.65	15.7	4.6	3	2.73	9.4	18.6	2	2.75	5.9	33.4
.75	5	3.38	11.4	8.7	3	3.44	6.9	27.0	2	3.46	4.3	44.9
.80	5	4.45	8.0	15.3	2	4.49	4.7	38.7	2	4.50	2.9	60.9
.85	4	6.18	5.1	25.6	2	6.19	3.2	56.9	1	6.20	1.9	85.7
.90	4	9.56	2.9	44.3	2	9.56	1.9	90.3	1	9.56	1.1	∞
.95	4	19.61	1.3	95.1	2	19.59	.8	∞	1	9.58	.5	∞

Table 2. Optimal Threshold Levels and System Performance.

	r = 0.20				r = 0.30				r = 0.40			
ρ	K_0	L_{K_0}	(1)	(2)	K_0	L_{K_0}	(1)	(2)	K_0	L_{K_0}	(1)	(2)
.05	3	.06	12.7	0.0	2	.07	6.1	0.0	1	.08	2.3	.3
.10	3	.14	18.7	0.0	2	.15	8.8	.2	1	.16	3.0	1.3
.15	3	.22	21.1	.1	1	.24	9.5	.9	1	.26	2.7	3.3
.20	3	.31	21.4	.3	1	.34	9.3	2.4	1	.36	2.1	6.2
.25	2	.43	20.3	.8	1	.46	8.5	4.8	1	.48	1.3	9.9
.30	2	.55	18.5	2.0	1	.59	7.4	8.0	1	.62	.4	14.6
.35	2	.70	16.3	3.8	1	.73	6.3	12.1	0	.76	0.0	20.5
.40	2	.86	14.0	6.5	1	.90	5.1	17.1	0	.92	0.0	27.4
.45	2	1.06	11.6	10.1	1	1.09	4.1	22.9	0	1.11	0.0	35.1
.50	2	1.28	9.3	14.6	1	1.30	3.1	29.8	0	1.32	0.0	43.6
.55	2	1.55	7.2	20.3	1	1.56	2.3	37.8	0	1.57	0.0	53.2
.60	1	1.86	5.8	27.7	1	1.87	1.6	47.2	0	1.87	0.0	64.4
.65	1	2.25	4.7	36.5	1	2.26	1.0	58.5	0	2.25	0.0	77.8
.70	1	2.76	3.6	47.5	1	2.77	.6	72.6	0	2.75	0.0	94.4
.75	1	3.46	2.7	61.6	1	3.47	.2	91.1	0	3.44	0.0	∞
.80	1	4.48	1.9	81.3	1	4.47	0.0	∞	0	4.46	0.0	∞
.85	1	6.18	1.3	∞	0	6.18	0.0	∞	0	6.14	0.0	∞
.90	1	9.54	.8	∞	0	9.53	0.0	∞	0	9.49	0.0	∞
.95	1	19.56	.3	∞	0	19.55	0.0	∞	0	19.51	0.0	∞

example, when the fast server is 20 times faster than the slow one ($r = 0.05$) and if ρ, say, is 0.8, a substantial reduction of queue length (and waiting time) can be achieved if the slow server is used in a correct way (i.e., allowing stalling). The practical lesson from this is that one should never discard an obsolete service device when new technology provides a much faster device. The slow server can be of value if properly used.

Finally, it is perhaps proper to note that the model presented here and represented by Figure 1 can be used to study situations when Server 1 is preferred over Server 2 for reasons other than speed--for example, because it is less expensive to use. Such will be the case, for example, when Server 1 is the "in house" computer while Server 2 is an outside computer which can be used for a fee. An optimization model for such situations can be developed using the results obtained in this paper.

4. On Social Versus Individual Customer Optimization

Studies of this topic began with the work of P. Naor [1969] and the latest results, in a most general setting, may be found in Bell and Stidham [1983]. They study a situation where customers have to make decisions on whether or not to join a queue, or which server to join, when the cost structure has a built-in tradeoff between a "reward" and a "cost". The reward (received by each customer who completes service) represents the value of receiving service and the cost (per unit of time) represents the value of time lost in waiting. The issue is whether customers seeking their own self optimization follow a decision rule which is socially optimal in the sense of maximizing average net reward (per customer) over all potential customers. It was shown that in general this is not the case and customers tend to join the queue more than is necessary for social optimality. However, a central controlling agency may, by levying tolls, create an environment in which

social optimality coincides with individual customer's optimality.

The present model provides another example of the same phenomenon with some added nice features. Here all customers join the queue and it is not necessary to introduce a special cost structure involving a "reward" that is difficult to measure. The natural "cost" is waiting time, or its value, and decisions are made on the basis of this "cost" only. On the other hand, since we do not have a closed form expression for $K_0 = K_0(r,\rho)$, our conclusions will be either qualitative or in terms of K_0.

For $n = 1,2,3,\ldots$ and $0 < r < 1$, let $\rho_n(r) = \sup\{\rho: L_n < L_{n-1}\}$ and $\rho_0(r) \equiv 1$. Thus, ρ_n is the lower boundary of the region, in the $r - \rho$ plane, where the socially optimal system is the one in which n customers stall. Now fix r and ρ and recall that a rational customer seeking self optimization joins the slow server if, and only if, the number of stalling customers is at least $1/r$. (His decision rule is, of course, independent of ρ.) This rule may or may not coincide with the socially optimal value K_0. Let

$$A_n = \{(r,\rho): 1/(n+2) \leq r \leq 1/(n+1),\ \rho \leq \rho_n(r)\}$$

$$B_n = \{(r,\rho): (r,\rho) \notin A_n,\ \rho_n(r) < \rho < \rho_{n+1}(r)\}$$

and set $A = \cup A_n$, $B = \cup B_n$. (These sets are shown in Figure 3.) Then if $(r,\rho) \in A$, social and individual optimality lead to the same decision rule; when $(r,\rho) \in B$ they differ. In the latter case customers seeking self optimization will stall more than is appropriate for social optimality. Applying the language of Bell and Sidham to our case, self-interested individuals tend to overcongest the fast server. But again, as in previous studies, a central agency in charge can, by levying tolls, create an

environment in which self-interested individuals will behave in a socially optimal way. The necessary toll is a charge for using the fast server, and its value is

$$\frac{\mu_1 - \mu_2 c K_0(\rho,r)}{\mu_1 \mu_2},$$

where c is the cost of waiting per time unit. This toll should be charged whenever $(\rho,r) \in B$.

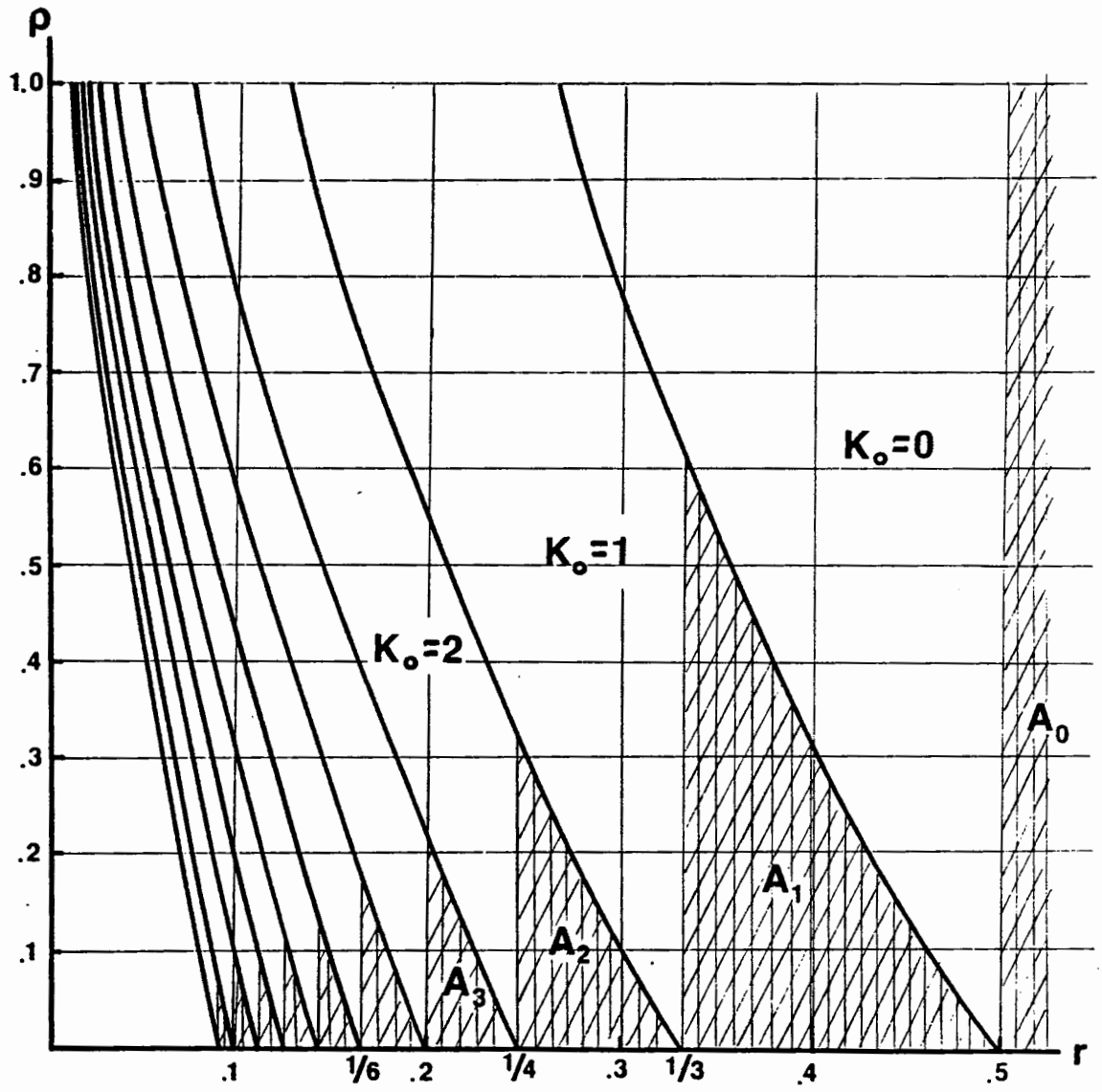

Figure 3. Individual Optimal Policies Versus Overall Optimal Policies

Acknowledgments: During the course of this investigation, I had some useful conversations with Erhan Çinlar. I am also grateful to Robert E. Machol for pointing out several references of which I was unaware.

References

Baily, T. J. 1957. Some Further Results in the Non-Equilibrium Theory of a Simple Queue. J. Roy. Stat. Soc. B19, 326-333.

Bell, C. E., S. Stidham. 1983. Individual Versus Social Optimization in the Allocation of Custoemrs to Alternative Servers. Mgmt. Science 29, 831-839.

Larsen, R. L. 1981. Control of Multiple Exponential Servers with Applications to Computer Systems. Computer Science Technical Report Series No. TR-1041, University of Maryland, College Park, Maryland.

Lin, W., P. R. Kumar. 1982. Optimal Control of a Queueing Sytem with Two Heterogeneous Servers. Mathematics Research Report No. 82-83, Department of Mathematics, University of Maryland, Baltimore County, Catonsville, Maryland.

Naor, P. 1969. On Regulation of Queue Size by Levying Tolls. Econometrica, 37, 15-24.

Prabhu, N. U. 1965a. Stochastic Processes, Macmillan, New York.

Prabhu, N. U. 1965b. Queues and Inventories, John Wiley, New York.

Rubinovitch, M. 1982. The Slow Server Problem. J. Appl. Prob. To appear.

Warland, J. 1983. A Note on "Optimal Control of a Queueing System with Two Heterogeneous Servers," Technical Report, Department of Electrical Engineering and Computer Sciences and Electronic Research Laboratory, University of California, Berkeley, California.

Appendix

Let T_{ij} be the first passage time from state i to state j (j > i) in an M/M/1 queue with arrival rate λ and service rate μ_1. Let $C_{ij}(\theta) = Ee^{-\theta T_{ij}}$. Then for j > i

$$(A.1) \qquad C_{ij}(\theta) = \frac{\lambda(z_1^{i+1} - z_2^{i+1}) - \mu_1(z_1^i - z_2^i)}{\lambda(z_1^{j+1} - z_2^{j+1}) - \mu_1(z_1^j - z_2^j)}$$

(Baily, 1957), where z_1 is the larger root and z_2 is the smaller root of

$$(A.2) \qquad \lambda z^2 - (\lambda + \mu_1 + \theta)z + \mu_1 = 0.$$

The expressions in (A.1) with j = K, i = K - 1, may thus be used in lieu of (4) if desired. Here we wish to show how (A.1) can easily be proven from (4).

To prove this let $C_{ij}(\theta)$ be formally defined by (A.1). Then $C_{01}(\theta) = \lambda/(\lambda + \theta)$. Also, from (A.2) $\lambda z_n^i = (\lambda + \mu_1 + \theta)z_n^i - \mu_1 z_n^{i-1}$ for n = 1,2. So, from (A.1)

$$C_{i-1,i}(\theta) = \lambda[\lambda + \mu_1 + \theta - \mu_1 C_{i-2,i-1}(\theta)]^{-1}.$$

Since this is the same as (4) it follows by induction that $C_{i-1,i}$, so defined, is the Laplace transform of $T_{i-1,i}$. Now, for j > i

$$T_{ij} = T_{i,i+1} * \cdots * T_{j-1,j}$$

and

$$C_{ij}(\theta) = Ee^{-\theta T_{ij}}.$$